Vulnerable University

Abhishek birdawade
Studenet, Department of computer science engineering
G H raisoni of college and engineering and management,pune
Pune, India
abhishek.birdawade.cs@ghrcem.raisoni.net

Srushti Deshpande
Student, Department of computer science engineering
G H raisoni of college and engineering and management,pune
Pune, India
srushti.deshpande.cs@ghrcem.raisoni.net

Vaibhav Wagh
Student, Department of computer science engineering
G H raisoni of college and engineering and management,pune
Pune, India
vaibhav.wagh.cs@ghrcem.raisoni.net

Gayatri Bedre
Teacher, Department of computer science engineering
G H raisoni of college and engineering and management,pune
Pune, India
gayatri.bedre.cs@ghrcem.raisoni.net

Abstract— **Identifying vulnerabilities and ensuring security functionality by security testing is a widely applied measure to evaluate and improve the security of software. Due to the openness of modern software based systems, applying appropriate security testing techniques is of growing importance and essential to perform effective and efficient security testing. Therefore, an overview of actual security testing techniques is of high value both for researchers to evaluate and refine the techniques and for practitioners to apply and disseminate them. For this purpose, it first summarizes the required background of testing and security engineering. Then, basics and recent developments of security testing techniques applied during the secure software development lifecycle, i.e., model-based security testing, code-based testing and static analysis, penetration testing and dynamic analysis, protection regression checking out are discussed. Finally, the safety checking out strategies are illustrated via way of means of adopting them for an instance three-tiered net-primarily based totally enterprise programs. All above cited stages are protected on this project. By making use of the project, the customers can be capin a position to check their abilities, analyze new abilities and audit real-global web sites with ease.**

Keywords— Cyber Security, Security testing, Penetration testing, Bug hunting, Authentication, Authorization.

I. Introduction

Now a days concepts like cloud computing, location-based services, and social networking, modern IT system are permanently connected to other systems and process highly sensitive data. These interconnected system are subject to security attacks that can lead to significant security incidents affecting the technical infrastructure or its environment. Exploited vulnerabilities can cause significant cost. Example, due to downtime or data changes. A high percentage of all software security incidents were caused by attackers exploiting known vulnerabilities. An important effective, and widely used tool for improving software security is security testing techniques that identify vulnerabilities and assure security functionality.

Software testing is concerned with evaluating software products and associated artifacts to determine whether they meet specific requirements, demonstrate fitness for purpose, and find bugs. Security testing verifies and verifies requirements of software systems in terms of security properties such as confidentiality, integrity, availability, authentication, authorization, and non-repudiation.

The vulnerable web applications can be used by web developers, security auditors, and penetration testers to practice their knowledge and skills during training sessions (and especially afterwards), as well as to test at any time the multiple hacking tools and offensive techniques available, in preparation for their next real-world engagement.

Vulnerable applications available to security professionals for hacking, offensive and defensive activities, so that they can manipulate realistic web environments without going to jail.

II. Literature Survey

1. *Security Testing: A Survey Michael Felderer, Matthias Büchler, Martin Johns (03/11/2015)*

   Remark:- This chapter provided a broad overview of recent security testing techniques

2. *A security testing platform for Wireless Sensor Networks Min Wei, Shuaidong Zhang, Keecheon Kim (10/03/2016)*

   Remark:- The test results show that the platform is a feasible platform for assessing device security levels in WSNs.

3. *Vulnerability Analysis Using the Interactive Application Security Testing (IAST) Approach for Government X Website Applications Hermawan Setiawan, Lytio Enggar Erlangga, Ido Baskoro (29/01/2001)*

   Remark:- Using the IAST approach, a total of 249 vulnerability risks were identified.

4. *Intelligent Testing in Software Industry Anushka Lal, Girish Kuma (03/11/2021)*

   Remark:- This study aims to investigate the approach of intelligent testing for improving software testability.

5. *Automated Cyber Security Testing Platform for Industrial Control System Andrew Hahn, Daniel R. Sandoval, Raymond E. Fasano, Christopher Lamb (23/10/2021)*

   Remark:- A platform that incorporates these three components to provide the most accurate representation of actual NPP networks and controllers is developed in this paper

III. OBJECTIVE

Vulnerable University is damn vulnerable application which any user can easily download from our GitHub page for free and run on their local system with single command.

This application can also be used using Docker file for platform independency and without having need to download and install dependencies to run the application.Our application consists of different technologies and their associated vulnerabilities for not only testing but also for learning and understanding different bugs.

1. To identify the threats in the web application, servers & websites.
2. To find potential vulnerabilities of the system and try to resolve those vulnerabilities.
3. To help in detecting to every possible security risk in the system.
4. Examine the impact occurs due to vulnerabilities.
5. To help the organizations and developer teams to resolving the security problem.
6. Improve the skills of the security professionals.
7. Introducing the various kind of hacking techniques without violating cyber laws.
8. Provide cyber education to students regarding technical and non-technical vulnerabilities.

IV. METHODOLOGY

Security testing verifies software system requirements related to the security properties of assets. This includes confidentially, integrity, availability, authentication, authorization, and non-repudiation. These security properties can be defined as follows:

1. *Confidentially:-*

   It is the assurance that information will not be disclosed to unauthorized persons, processes, or devices.

2. *Integrity:-*

   Integrity exists if the data has not been tampered with from its source and has not been altered or destroyed either accidentally or maliciously.

3. *Availability:-*

   It ensures authorized users timely and reliable access to data and information services.

4. *Authentication:-*

   It is a security measure designed to verify the validity of a transmission, message, or originator, or to verify an individual's entitlement to receiver a particular category of information.

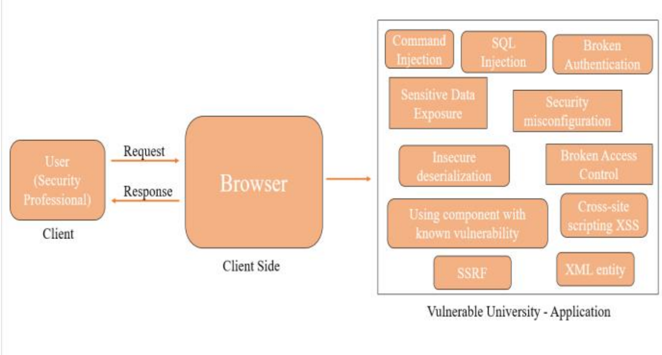
5. *Authorization:-*

   It provides authorized access rights to users, programs, or processes.

6. *Non.-repudiation:-*

   It is the warrantly that none of the parties involved in a transaction can subsequently dispute their involvement.

A. System Architecture Diagram:-



B. Technology Used:-

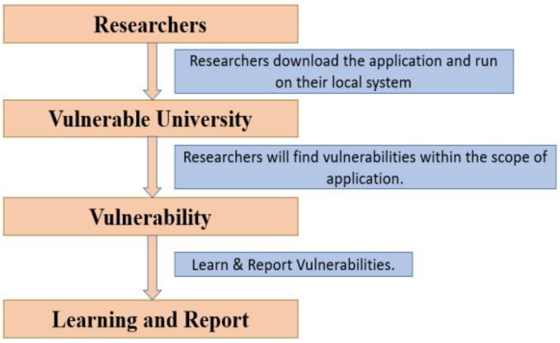
1. *Django:-*

   Django is free, open source and written in python. Django makes it easy to build application in python, so use Django to design your application. Django emphasizes component reusability, also known as DRY(Don't Repeat Yourself), with out-of-the-box features such as a login system, database connectivity, and CRUD(Create Read Update Delete) operation.

2. *Docker:-*

   Docker is a suite of Platform-as-a-Service product that use OS-level virtualization to deploy software in package called containers. The service has both free and premium tiers. The software that hosts containers is called the Docker engine. Using this technology makes the application platform esier to use.

C. Block Diagram:-


Researchers
Researchers download the application and run on their local system
Vulnerable University
Researchers will find vulnerabilities within the scope of application.
Vulnerability
Learn & Report Vulnerabilities.
Learning and Report

D. Planning:-

Vulnerable university is a single platform that is developed for security professionals to enhance and practice their skills.

1)Vulnerabilities Collection:-

We collected most of the common vulnerabilities both technical and non-technical & gathered the knowledge of how the vulnerabilities can be introduced in a web application and the PHP server.

2) Website designing:-
After listing the vulnerabilities we prepared how we can make our server and web application vulnerable to those vulnerabilities which the student can reproduce locally.

3) Creating the frontend:-
Then we created the vulnerable frontend interface of the website which the students can interact with and practice testing the vulnerabilities.

4) Creating the backend:-
After creating frontend of the platform, we created the backend of the application using PHP server which is intentionally vulnerable and cable of being deployed locally on any machine.

5) API middleware:-
After the frontend and backend both being created, we used APIs to connect them together using RESTful APIs, which also are vulnerable to CRUD related vulnerabilities and various other vulnerabilities.

6) Docker container:-
After creating the intentionally vulnerable university's web application with backend, we stacked it together in a docker container so that the student can easily deploy and run the web-application & start testing/practicing their hacking skills for good.

V. Conclusion

Cyber security is the method of protecting computer devices, networks and programs from cyberattacks. Cyber security is becoming more and more important. Cybercriminals are becoming more sophisticated, changing their attack methods and damaging an organization's image, creating very important conditions for an organization**.**

Vulnerable University helps improve the skills of security professionals and this application helps raise awareness of ethical hacking among students. Anyone can easily access this free application and run it locally**.** Students can learn the necessary skills and earn a good amount of money by finding bugs that work for many organizations on various platforms.
Organization**.**

VI. References

[1] Adam Doupe and Marco Cova and Giovanni Vigna. Why johnny can't pentest: An analysis of black-box web vulnerability scanners. In DIMVA 2010, 2010.

[2] N. Ayewah, D. Hovemeyer, J. D. Morgenthaler, J. Penix, and W. Pugh. Experiences using static analysis to find bugs. IEEE Software, 25:22–29, 2008. Special issue on software development tools, September/October (25:5).

[3] R. Bachmann and A. D. Brucker. Developing secure software: A holistic approach to security testing. Datenschutz und Datensicherheit (DuD), 38(4):257–261, apr 2014.

[4] R. Bachmann and A. D. Brucker. Developing secure software: A holistic approach to security testing. Datenschutz und Datensicherheit (DuD), 38(4):257–261, apr 2014.

[5] P. Ammann and J. Offutt. Introduction to Software Testing. Cambridge University Press, Cambridge, UK, 2008.

[6] J. Grossman, R. Hansen, P. Petkov, and A. Rager. Cross Site Scripting Attacks: XSS Exploits and Defense. Syngress, 2007.

[7] M. Anisetti, C. Ardagna, and E. Damiani. A low-cost security certification scheme for evolving services. In Web Services (ICWS), 2012 IEEE 19th International Conference on, pages 122–129, June 2012.

[8] B. Arkin, S. Stender, and G. McGraw. Software penetration testing. Security & Privacy, IEEE, 3(1):84–87, 2005.

[9] J. Bau, E. Bursztein, D. Gupta, and J. Mitchell. State of the art: Automated black-box web application vulnerability testing. In Security and Privacy (SP), 2010 IEEE Symposium on, pages 332–345. IEEE, 2010.

[10] A. D. Brucker, L. Br¨ugger, and B. Wolff. Formal firewall conformance testing: An application of test and proof techniques. Software Testing, Verification & Reliability (STVR), 25(1):34–71, 2015.